

Encoding dynamics for multiscale community detection: Markov time sweeping for the map equation

Michael T. Schaub,^{1,2,*} Renaud Lambiotte,³ and Mauricio Barahona^{1,†}

¹*Department of Mathematics, Imperial College London, London SW7 2AZ, United Kingdom*

²*Department of Chemistry, Imperial College London, London SW7 2AZ, United Kingdom*

³*Department of Mathematics and Naxys, University of Namur, 5000 Namur, Belgium*

(Dated: August 23, 2012)

The detection of community structure in networks is intimately related to finding a concise description of the network in terms of its modules. This notion has been recently exploited by the map equation formalism (M. Rosvall and C. T. Bergstrom, *PNAS*, **105**(4), pp. 1118–1123, 2008) through an information-theoretic description of the process of coding inter- and intra-community transitions of a random walker in the network at stationarity. However, a thorough study of the relationship between the full Markov dynamics and the coding mechanism is still lacking. We show here that the original map coding scheme, which is both block-averaged and one-step, neglects the internal structure of the communities and introduces an upper scale, the ‘field-of-view’ limit, in the communities it can detect. As a consequence, map is well tuned to detect clique-like communities but can lead to undesirable overpartitioning when communities are far from clique-like. We show that a signature of this behavior is a large compression gap: the map description length is far from its ideal limit. To address this issue, we propose a simple dynamic approach that introduces time explicitly into the map coding through the analysis of the weighted adjacency matrix of the time-dependent multistep transition matrix of the Markov process. The resulting Markov time sweeping induces a dynamical zooming across scales that can reveal (potentially multiscale) community structure above the field-of-view limit, with the relevant partitions indicated by a small compression gap.

I. INTRODUCTION

The analysis of biological, technical and social networks has become extremely popular in recent years [1–3]. The availability of high dimensional relational data coupled with increasing computational power has set the ground for the investigation of complex systems from a network perspective, i.e., each agent or entity is viewed as a node interacting via multiple links with other nodes in the network. Such a viewpoint aims to understand the global emergent behavior of the system from the interactions between the individual components of the system, in contrast to focusing on each part on its own.

In many cases of interest, complex networks are far from being unstructured and contain relevant subgroupings or *communities*, possibly organized into (not necessarily hierarchical) multiple levels [4]. The detection of such community structure can be of importance for the understanding of the interplay between the structural and functional features of the network. In particular, parts of the system operating on given scales could be represented with a simplified description at an appropriate level of coarse graining.

Community detection methods based on a variety of heuristics (including modularity [5, 6] and spectral partitioning methods [7–10] among many others—see Refs. [1, 11] for recent reviews) have been proposed to find an optimized split into communities. The communities thus found result from identifying groups with high

intra-community weights as compared to the expected weights in surrogate models of the network. In adopting such a structural criterion, these methods introduce an intrinsic scale that establishes limits on the communities they can detect, thus leading to potential mis-detection [12]. Furthermore, such single scale methods are not suitable for the analysis of networks in which there is not a single ‘best’ mesoscopic level of description, but rather multiple levels associated with different scales in the system [13].

In order to account for the presence of multiple levels of organization, multiscale methods have been introduced that allow to search for the right scale at which the network should be analyzed [14–17]. Recently, it has been shown that one can use the time evolution of a Markov process on the graph to reveal relevant communities at different scales in a process of dynamic zooming through the so-called *partition stability* [12, 18, 19]. As the Markov time increases, the diffusive process involves multistep transitions and explores further afield the structure of the graph, resulting in the detection of community structure across scales, from finer to coarser. This dynamic approach has the advantage that it provides a unifying framework for structural community detection methods (such as modularity and spectral methods), which can be seen as particular cases of this approach involving one-step measures.

A different perspective is provided by an information theoretic framework that considers the problem of finding communities in a network as a coding or compression problem [20–23]. The underlying idea is that the presence of communities should imply the existence of an efficient and concise way to encode the behavior of

* E-mail: michael.schaub09@imperial.ac.uk

† E-mail: m.barahona@imperial.ac.uk

a system in terms of its subgroups. Recently, the map equation method by Rosvall et al [21, 22] relies on a compression of the description length of a random walk inside and between communities to find good graph partitions. This method has received a lot of attention, since it has been shown to be extremely efficient on benchmark tests [24] outperforming the popular modularity [5, 6]. It has also been shown to be immune to the resolution limit [25] that affects the performance of modularity. However, the mathematical properties and possible limitations of the map equation remain relatively unexplored.

Here we show that the map equation can also be understood as a one-step method and, consequently, it suffers from an upper scale (the field-of-view limit) above which it cannot detect communities [12]. This limited field-of-view can lead to overpartitioning when communities are far from being clique-like [12]. In addition, the one-step map coding scheme also neglects the internal structure of the communities and, in doing so, introduces a bias towards communities that are locally fast mixing (and in this sense clique-like). We also show that the quality of the map partitioning can be assessed through the existence of a small compression gap, i.e., a small distance between the compression achieved by Map and its theoretical limit given by the true entropy rate of the Markov process. To alleviate some of these limitations, we introduce a dynamical approach that introduces time explicitly into the map coding scheme, by considering the time-dependent multi-step transition matrix of the Markov process on the network as the object of the map encoding. This introduces a dynamic zooming by sweeping through the Markov time, which allows the detection of multiscale community structure with the map equation formalism.

II. COMMUNITY DETECTION FROM A CODING PERSPECTIVE: THE MAP EQUATION

The map formalism considers the problem of partitioning a network into non-overlapping communities from a coding perspective. The original map formalism [21] equates the quality of the partition to the efficiency of a code that would describe the notional transitions of a random walker inside and between communities. The Infomap algorithm can then be used to obtain good partitions through the optimization of this quality function. The underlying principle is that the code for such *one-step* transitions of the random walker can be efficiently compressed in the presence of a strong community structure: short names for nodes (codewords) can be reused in different communities, much like street names can be reused in different cities of a country [21]. In the original map equation, the movement of the walker is described in terms of two kinds of codebook. The first kind of codebook is specific to each community and assigns a unique codeword for each node inside it and a particular exit codeword for the community. An additional codebook

contains unique codewords that describe the movements between different communities. More recently, a hierarchical extension of the map formalism (a recursive version of the original method) has been presented [22] as well as an extension for overlapping modules [26]. We do not consider these extensions in detail here, as both methods are based on the same principles of the standard map equation and our findings are applicable to these as well.

A. Definitions and notation

An explicit rewriting of the original map formalism in terms of the stationary distribution of a random walk is as follows. Consider a discrete time Markov process on a graph with N nodes:

$$\mathbf{p}_{k+1} = \mathbf{p}_k D^{-1} A \equiv \mathbf{p}_k M, \quad (1)$$

where \mathbf{p}_k is the $1 \times N$ (node) probability vector, A is the (weighted) adjacency matrix of the graph, D is the diagonal matrix containing the (weighted) degree of each node, and we have also defined M , the transition matrix of the random walk. The stationary distribution of the random walk, π , is then given by:

$$\pi = \pi M. \quad (2)$$

Consider now a partition of the network into c communities indexed by $\alpha = 1, \dots, c$. At stationarity, the probability of leaving community α (or of arriving at community α) is

$$q_{\alpha\curvearrowright} = \sum_{i \in \alpha} \sum_{j \notin \alpha} \pi_i M_{ij},$$

and the overall probability of changing community is

$$q_{\curvearrowright} = \sum_{\alpha=1}^c q_{\alpha\curvearrowright}.$$

Similarly, the probability to stay within or to leave community α is

$$p_{\curvearrowright}^{\alpha} = q_{\alpha\curvearrowright} + \sum_{i \in \alpha} \pi_i.$$

The map equation then defines the per-step description length of a code associated with this partition as:

$$L_M = \sum_{\alpha=1}^c p_{\curvearrowright}^{\alpha} H(\mathcal{P}^{\alpha}) + q_{\curvearrowright} H(\mathcal{Q}), \quad (3)$$

a weighted combination of the Shannon entropies:

$$H(\mathcal{P}^{\alpha}) = -\frac{q_{\alpha\curvearrowright}}{p_{\curvearrowright}^{\alpha}} \log_2 \left(\frac{q_{\alpha\curvearrowright}}{p_{\curvearrowright}^{\alpha}} \right) - \sum_{i \in \alpha} \frac{\pi_i}{p_{\curvearrowright}^{\alpha}} \log_2 \left(\frac{\pi_i}{p_{\curvearrowright}^{\alpha}} \right)$$

$$H(\mathcal{Q}) = -\sum_{\alpha=1}^c \frac{q_{\alpha\curvearrowright}}{q_{\curvearrowright}} \log_2 \left(\frac{q_{\alpha\curvearrowright}}{q_{\curvearrowright}} \right).$$

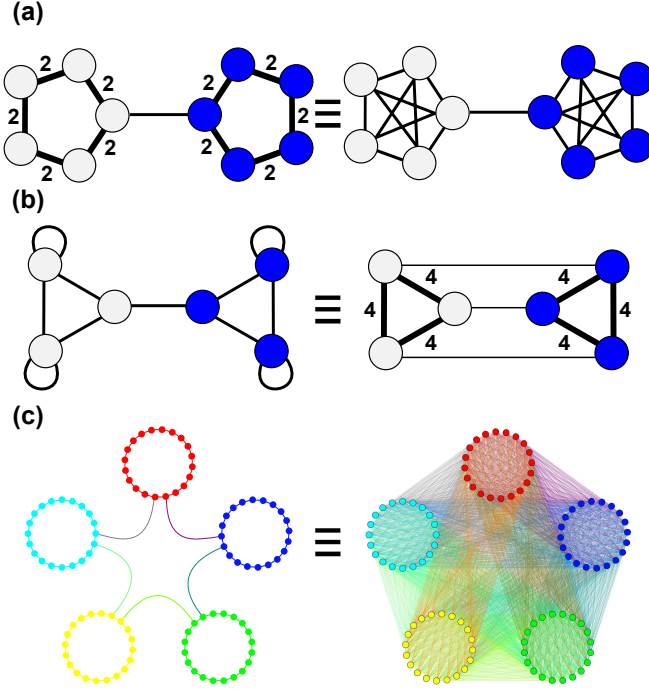


Figure 1. (Color online) **Equivalent graph partitions for the map equation.** Because the map equation ignores the specific connectivity of the graph, graph partitions with equal equilibrium and leaving probabilities become indistinguishable to Map. Different communities are represented by different colors (shades of gray). Unless indicated, the weight of the edge is 1. (a) Two graphs with different intra-community connectivity but the same map coding length, L_M . (b) Two graphs with different inter-community connectivity and the same L_M . (c) Two graphs with equal L_M but very different inter- and intra-community connectivity. From the viewpoint of Map, a ring-of-rings is equivalent to a clique-of-cliques with different weights.

The two terms in Eq. (3) correspond to two classes of codebooks that encode one-step transitions at stationarity viewed through the prism of the given partition. The first term stems from the “community-centric” codebooks with probability distributions \mathcal{P}^α (and associated entropy) of being at or leaving from each of the communities. The second term corresponds to the “inter-community” codebook with distribution \mathcal{Q} (and associated entropy) of changing community.

In the original map formalism it is proposed that a low L_M is a characteristic of good partitions and the Infomap algorithm is used to search computationally for partitions with low L_M .

III. MAP ENCODES BLOCK-AVERAGED, ONE-STEP TRANSITIONS: IMPLICATIONS FOR COMMUNITY DETECTION

As shown by the definitions above, the original map equation does not fully code for the dynamics of the

Markov process (1), as it only uses quantities derived from block-averaging of one-step transitions at stationarity. The simplifications involved in block-averaging the structure and in ignoring longer-term dynamics both have inter-related implications for community detection, which we now study in detail.

A. Block-averaging the connectivity: the compression gap and a bias towards over-fitting to clique-like communities

An examination of the terms in the map equation (3) reveals that the implicit block-averaging neglects the internal structure of the communities as well as the detailed inter-community connectivity. More precisely, given a particular partition, all graphs with the same equilibrium distribution π and overall leaving probabilities $q_{\alpha\sim}$ will be indistinguishable in terms of their map quality, L_M , as exemplified in Figure 1.

From the viewpoint of entropies, the map equation (3) is formally equivalent to a weighted sum of the entropies of i.i.d. stochastic processes with states visited according to: normalized “community-centric” probabilities $\{\{\pi_i/p_{\alpha}^\alpha\}_{i \in \alpha}, q_{\alpha\sim}/p_{\alpha}^\alpha\}_{\alpha=1}^c$; and normalized “leaving” probabilities $\{q_{\alpha\sim}/q_{\sim}\}_{\alpha=1}^c$, respectively. Alternatively, this procedure may be seen as formally equivalent to using a block-averaged transition matrix corresponding to a block-structured weighted (and in general directed) *complete graph* with self-loops. Consequently, Infomap exhibits a bias towards identifying communities that are formally equivalent to clique-like subgraphs.

In this sense, the map equation can be seen to code for a two-level, mean-field organization: one inside communities, one across communities. Such block-structured, all-to-all models are a good representation of community structure based on hierarchical cliques-of-cliques. Indeed, the map equation performs well in block-structured Erdős-Renyi benchmarks [24] and is not afflicted by the ‘resolution limit’ [25]. On the other hand, there are important networks with a more marked local structure in which communities are not clique-like [12]. Because the map formalism has not been designed to detect such non clique-like communities with large effective distances, Infomap will tend to overpartition such networks.

Ignoring the detailed connectivity: the compression gap of the map equation

The fact that Map ignores the detailed connectivity inside and outside the communities leads to a sub-optimal coding scheme. This sub-optimality can be quantified through the *compression gap* (defined below), which can be used as a measure of when the Map block-averaging assumptions are a valid simplification for the network under study.

Consider the Markov chain with transition matrix M

and stationary distribution π , as given in Eq. (1). The most efficient coding of the dynamics of the associated Markov process at stationarity is bounded from below by the entropy rate h [27, 28]:

$$h(\pi; M) = - \sum_{ij} \pi_i M_{ij} \log_2(M_{ij}). \quad (4)$$

The corresponding optimal encoding can be asymptotically achieved by endowing each node with a dictionary for its outgoing links, as shown by Shannon [27]. This is a kind of ‘edge encoding.’

On the other hand, if we consider a coding scheme that gives each node a unique name within the whole graph, (i.e., a ‘node encoding’), then the corresponding coding length is bounded by the entropy rate of the i.i.d. random variable with probability distribution π , which is equal to the entropy of the stationary distribution:

$$H(\pi) = - \sum_i \pi_i \log_2(\pi_i). \quad (5)$$

The map coding scheme can be seen as a mixture of both: it encodes nodes uniquely within communities, but encodes for transitions (‘edges’) between communities. Therefore, in general,

$$h(\pi; M) < L_M \leq H(\pi), \quad (6)$$

and Map is sub-optimal in terms of its coding length [29], as recognized by Rosvall and Bergstrom in their original publication [21]. This sub-optimality can be understood with a simple example: consider a community α_{out} from which there is only one possible link to another community α_{to} . Map encodes this transition with two codewords: an exit codeword to signal the leaving of α_{out} and a codeword to identify the destination community α_{to} . Clearly, the second codeword is redundant.

Importantly, if the graph is a weighted, directed clique (i.e., with transition matrix $M = \mathbf{1}\pi$), then $h(\pi; M) = H(\pi)$ and the two coding schemes (‘edge’ and ‘node’) give the same result (see Figure 2(a)). Therefore, the sub-optimality of the map coding is minimal when the graph is close to a clique. Consequently, the minimization of the map cost function is well suited to identify community structure that is a clique of cliques: within each community Map uses a ‘node’ encoding while between communities Map encodes transitions by default. In such a scenario, the map coding scheme is nearly optimal and close to the entropy rate.

The sub-optimality of the map encoding plays a significant role when encoding communities with restricted connectivity. For instance, if the community is a ring, a random walker has only two possible nodes to transition to, instead of n_α as assumed by Map. In this case, there is a large gap between the map description length L_M and the optimal limit established by $h(\pi; M)$, indicating that the full consideration of the graph structure in the Markov dynamics could be exploited for a better encoding (see Figure 2(b) for an example).

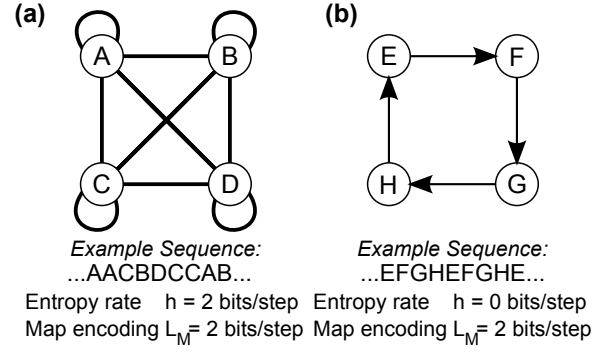


Figure 2. **The compression gap of the map coding scheme.** (a) For a clique with self loops, the map coding scheme is optimal (assuming no communities) and is equivalent to a uniform i.i.d. process with four states. (b) In a directed cycle, the map coding is far from optimal. The movement of a random walker on this graph can be encoded by just denoting the starting position but the map coding scheme enforces unique names for each node and thus requires at least 2 bits/step.

This discussion highlights the fact that the block-averaging implicit in the original map scheme leads to a sub-optimality of the proposed map coding scheme that becomes significant when the network cannot be well described as a clique-of-cliques. In order to quantify this effect, we define the *compression gap*, δ :

$$\delta = (L_M - h)/h, \quad (7)$$

which measures how close the map encoding is to optimality. Note that other measures for the compression gap, such as $\delta' = (L_M - h)/(H - h)$, could be used and may be more suitable, or sensitive, in some cases. In this manuscript we stick mostly to the slightly simpler expression of δ , as it is sufficient for our purposes. The compression gap can be used to establish when the communities identified by Map are far from being clique-like and hence serves as an indicator of the reliability of the partitions obtained by Infomap, as shown below.

B. One-step transitions: the field-of-view limit and a bias towards overpartitioning of non clique-like communities

As discussed above, the original map formalism is based on an implicit clique-like concept of community, and a community structure as a (statistical) clique of cliques. Although this model has proved successful in a variety of fields [24, 30], relevant technological, social and biological networks are far from being clique-like [12]. In such cases, Infomap might tend to overpartition communities as a result of an upper scale (the ‘field-of-view limit’) which stems from map encoding only for one-step transitions at stationarity. This field-of-view limit affects all one-step methods, including not only Map but

also modularity. The field-of-view occurs on the opposite end of the well-known resolution limit that appears as a lower scale for modularity [25] but does not seem to impact Map [17].

Overpartitioning of lattice-like graphs

The overpartitioning induced by the field-of-view limit can be understood analytically through the following simple examples of lattice-like graphs.

First, consider a cycle graph of length N with unweighted edges. The equilibrium distribution of the random walk on this graph is $\pi_i = 1/N, i = 1, \dots, N$. This graph has no community structure and the only relevant partition should be the global “all-in-one”.

For a partition of the ring into $c \geq 2$ communities indexed by α we have:

$$\{q_{\alpha\sim} = 1/N, \forall \alpha; \quad q_{\sim} = c/N; \quad p_{\odot}^{\alpha} = (n_{\alpha} + 1)/N\}$$

where n_{α} is the number of nodes in community α and, clearly, $\sum_{\alpha=1}^c n_{\alpha} = N$. The map cost function of this partition is

$$L_M(\{n_{\alpha}\}_{\alpha=1}^c) = \frac{c}{N} \log_2(c) + \sum_{\alpha=1}^c \frac{n_{\alpha} + 1}{N} \log_2(n_{\alpha} + 1). \quad (8)$$

Using convexity arguments, it is easy to show that for a given N and c , the minimal L_M is attained for the partition with equally-sized communities with $n_{\alpha} = N/c, \forall \alpha$, if it exists. For such a partition, the map equation (8) becomes:

$$L_M(\{N/c\}_{\alpha=1}^c) = \left(1 + \frac{c}{N}\right) \log_2(N + c) - \log_2(c), \quad (9)$$

with $c \geq 2$. The case $c = 1$ is the trivial “all-in-one partition” with $L_M(\{N\}_{\alpha=1}) = \log_2 N$.

The relevant Map optimization for the cycle graph of size N is then equivalent to finding which of the equal partitions into c communities has the lowest L_M :

$$\min_c L_M(\{N/c\}_{\alpha=1}^c).$$

Assume N/c to be real to facilitate the analysis, a relaxation which our numerics show not to affect the result. Then the partition with minimal L_M has equal communities of size N/c^* with $c^*(N)$ given by:

$$\ln(N + c^*) = \frac{N}{c^*} - 1 \quad c^* \geq 2. \quad (10)$$

It is easy to show that, for a long enough ring, such a partition will have lower L_M than the ‘all-in-one’ partition. Indeed, the map equation partitions all cycles with $N \geq 10$.

Similar results are obtained for the regular k -cycles used as the starting point for the small-world construction (see Section V C). In this case, Map partitions the

k -cycle into equally-sized communities of size N/c^* given by:

$$\ln\left(\frac{2N}{k+1} + c^*\right) = \frac{2N}{c^*(k+1)} - 1 \quad c^* \geq 2. \quad (11)$$

The same reasoning can be applied to a torus network, i.e., the cartesian product of two cycles of lengths R and r , with $N = rR$. This graph can be thought of as the discretization of a 2-dimensional lattice with periodic boundary conditions. It is easy to show that the optimal *radially symmetric* partition of the graph (with $R > r$) is into communities of size N/c^* with c^* given by:

$$\ln(2R + c^*) = \frac{2R}{c^*} - 1 \quad (12)$$

Therefore, as the size of the lattice N increases, Infomap will partition the torus into smaller sections. Our numerical exploration shows that the above solution is a conservative estimate and the overpartitioning induced by Map is even more acute for the torus: As N grows, other even smaller patch-like partitions are obtained by the Infomap optimization.

IV. A DYNAMICAL ENHANCEMENT OF THE MAP SCHEME: MARKOV TIME SWEEPING FOR THE MAP EQUATION

As discussed above, the original map equation does not fully account for the dynamics of the Markov process (1), as it only uses quantities derived from block-averaged one-step transitions. Such a simplification is reasonable for clique-like communities, which exhibit a small compression gap and can be fully explored in one step. However, networks of interest sometimes possess a multi-scale, non clique-like community structure which will go unrecognized by the original map equation due to its intrinsic bias towards cliques and the ensuing field-of-view limit.

The limitations of the map equation in such scenarios can be overcome by adopting concepts from *partition stability*, a recently introduced dynamical framework for community detection [12, 18, 19]. The idea is to consider the time evolution of the Markov process as a means to unfolding systematically the graph structure at different scales. This *Markov time sweeping*, which is equivalent to considering multi-step transitions, applies a natural zooming process (from small to large scales) to the network. A key aspect of this approach is the systematic sweeping across scales provided by the dynamics, which minimizes the effects of the resolution and field-of-view limits. For an extended discussion, see [12, 18, 19, 31].

This Markov time sweeping can be used to endow the map equation with a dynamic zooming that allows it to detect multi-scale community structure, with relevant partitions characterized by a low compression gap. For simplicity, consider the continuous version of the Markov

process (1) associated with a graph with adjacency matrix A on N nodes:

$$\dot{\mathbf{p}} = -\mathbf{p} D^{-1} L, \quad (13)$$

where \mathbf{p} is a $1 \times N$ vector of probabilities, D is the diagonal matrix containing the weights of each node and $L = D - A$ is the graph Laplacian. It is easily verified that this continuous-time Markov process has the same stationary distribution as the discrete-time random walk (1) [18, 19].

The analytical solution of this system leads us to consider the discrete-time process:

$$\mathbf{p}_{k+1} = \mathbf{p}_k T(t), \quad (14)$$

where $T_{ij}(t) = [e^{-tD^{-1}L}]_{ij}$ is the effective transition probability between nodes i and j after a (Markov) time t . Within this framework, it is easy to see that the original map formulation considers the *linearized* version of $T(t)$ evaluated at time $t = 1$. Consequently, the original map equation scheme is included as a particular case in our formulation and we can always recover the standard Map results under our scheme [32].

Our approach is then to use the map equation to analyze the community structure of the *time-dependent weighted network* $DT(t)$ as a function of the (Markov) time, t . As time grows, the transition matrix T becomes less sparse and more clique-like, yet in a structured manner that reflects the community structure of the network [19]. Consequently, the leaving probabilities $q_{\beta \sim}(t) = \sum_{i \in \alpha} \sum_{j \notin \alpha} \pi_i T_{ij}(t)$ increase with increasing time; the cost for encoding distinct communities increases too; and map tends to find coarser communities that can be better represented as cliques. More specifically:

- For $t \rightarrow 0$, the leaving probabilities go to zero and the map equation is minimized by setting each node in its own community, as can be easily verified.
- For $t \rightarrow \infty$, we approach the limit of an i.i.d. random process, i.e., $T(t) \rightarrow \mathbf{1}\pi$, where $\mathbf{1}$ is the vector of ones. In this limit, the map encoding for the “all-in-one” partition is optimal, since it results in a description length which is equivalent to the entropy rate. More precisely, it is easy to see from Eq. (6) that $\delta(t) \rightarrow 0$ as $t \rightarrow \infty$.
- For intermediate times, the Markov time acts as a natural resolution parameter and the partitions of the time-dependent weighted graph $DT(t)$ become increasingly coarser. By following the time evolution, we can check whether a particular partition corresponds merely to a transient or whether it is persistent for a range of times.

Furthermore, the compression gap (7) can be used as an information-theoretic indicator of the reliability of the partitions found by Infomap at different Markov times. As discussed above, a low δ is expected when the partition reflects a community structure close to that of a

clique of cliques, thus conforming to the assumptions underlying the map formalism. Therefore, low values of $\delta(t)$ can be used to indicate relevant map partitions and also to identify the existence of a multi-scale community structure in the network.

This Markov time sweeping brings to the map equation what the *partition stability* offers to modularity [18, 19]; namely, the possibility to use time as a means to scan naturally through the resolution of community detection (from fine to coarse) in a manner that is consistent with the Markov dynamics on the graph. From this dynamical viewpoint, the standard map equation corresponds to a time-snapshot of the diffusion dynamics. Furthermore, this dynamical approach is a natural framework for the map scheme, since it introduces a time-dependent but finite probability of jumping from any node to any other node at all times, in line with the formalism underpinning the map equation.

V. SOME ILLUSTRATIVE EXAMPLES

In this section, we illustrate the use of *Markov sweeping map* with simple examples. The procedure is as follows: For each Markov time, we construct the time-dependent network defined by $DT(t)$. We then optimize the (time-dependent) map cost function using the implementation of Infomap for directed graphs found online at <http://www.tp.umu.se/~rosvall/>, slightly modified to enable self-loops in the graphs. We only consider here undirected networks but the method can be extended easily to directed graphs when we allow for teleportation [33]. For all examples, 100 runs of the Infomap algorithm at each Markov time were used to find the optimal partition.

A. A network without community structure: the cycle graph

As a first example, we apply Markov time sweeping to the ring network discussed in Section III B. Recall that for the cycle graph with $N = 20$ nodes, our analytical arguments show that the original map scheme leads to a non-intuitive partition into 5 equal communities, instead of the expected ‘all-in-one’ partition. However, the high compression gap of the 5-way partition found by the standard Map ($\delta \approx 2.48$) confirms that this partition is far from being formed by clique-like communities. Because standard Map is being applied to a network which does not conform to the implicit assumptions about community detection in the original map framework, we see an overpartitioning in this case.

As seen in Figure 3, analyzing this network with the Markov-sweeping version of Map reveals that there is no significant community structure in this graph. Only the singleton partition (at very short times) and the global partition (at very long times) provide significant group-

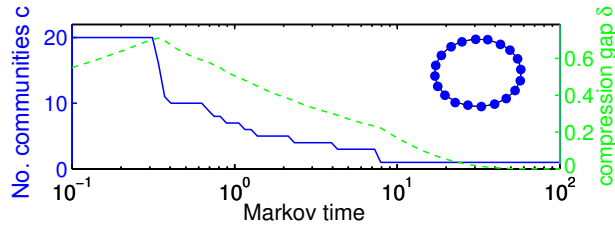


Figure 3. (Color online) **Markov sweeping map for a cycle graph with $N = 20$.** As the Markov time increases, the map partitioning goes from the finest possible partition to the global ‘all-in-one’ partition (solid blue line). However, as indicated by the featureless decay of the compression gap δ with no clear minima (dashed green line), no other relevant community is found between those two extreme partitions, thus signaling the lack of community structure. In this case, optimizing the standard map equation finds 5 communities but a large compression gap $\delta \approx 2.48$ indicates that this partition is unreliable. Inset: analyzed graph.

ings of the nodes while all other partitions show high values of δ .

B. A simple network with multi-scale community structure

Consider now a weighted graph with a distinct hierarchical community structure: two triangles of triangles with weighted links to reinforce the hierarchy (see inset of Figure 4). In this example, the standard map equation method identifies the fine structure of six small triangles. (We note that the hierarchical map equation uncovers the two-tier hierarchy of communities in this graph.)

Our proposed Markov sweeping map also recovers the hierarchy of partitions across time-scales, as indicated by the sharp decreases in the compression gap δ when the six-fold and the two-fold partition are detected (Figure 4). Our method also indicates over which timescales the relevant partitions appear to be natural. For instance, a change in the weights would induce changes in the lengths of the plateaux corresponding to the different levels of the hierarchy. As stated above, hierarchical Map is able to resolve this clique-like community structure (while standard Map finds only the fine structure). However, if the multi-scale structure is not clique-like, hierarchical Map may fail to resolve the multi-scale structure, as shown in the network of small-world communities discussed in the next section.

C. A ring of small-world communities

As a next scenario, we study a ring of five weakly connected small-world graphs [34] of 200 nodes each, as introduced in [12] (see Figure 5(a)). We use the CONTEST toolbox [35] to generate small-world communities

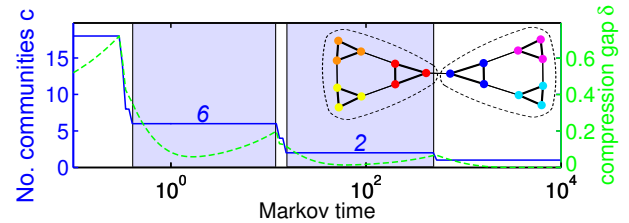


Figure 4. (Color online) **Markov sweeping map for a graph with a hierarchical community structure.** The graph analyzed (inset) has a clear community structure given by a hierarchy of triangles: the six smaller triangles (denoted by different colors) have edges within them of weight 100; they are grouped into two larger triangles with weaker links (the edges between the 6 small triangles have weight 10); the edge between the two big triangular structures has weight 1. The compression gap (dashed green line) shows two clear minima, indicating well defined partitions into 6 and 2 communities, corresponding to the two tiers of the hierarchy. Standard Infomap finds only the 6 small triangles ($\delta \approx 0.62$).

by adding random connections à la Newman-Watts [36] starting from a pristine world with two nearest neighbours [37] but allowing for the possibility of multiple shortcuts at each node.

As discussed in Section III B, the standard map equation will tend to overpartition lattice-like structures, such as the pristine worlds (k -cycles) used as starting point for the small-world construction. As given by Eq. (11), standard Infomap partitions the pristine world with $N = 200$ and $k = 2$ into 22 equally-sized communities.

This overpartitioning persists when few random shortcuts are added, as shown in Figure 5(b). Only when the average number of added shortcuts per node, here denoted by s , is greater than 3.5 (and the mean distance within the small-world has become small) does standard Map obtain the right split into five communities. This is consistent with our discussion pertaining the field-of-view, i.e., the smaller the mean path length, the more clique-like the structure. In this case, hierarchical Infomap can even give a non-intuitive partition into 4 communities, due to the non clique-like nature of the communities. On the other hand, Figure 5(c) shows that Markov sweeping allows Map to detect the relevant partition into 5 communities over an extended time-scale with a small compression gap.

VI. DISCUSSION

A key insight to emerge from the map equation formalism is the fact that a coarse-grained description of a graph in terms of its communities is intimately related to finding concise descriptions of the information flow on these networks, and hence to the field of coding theory and data compression. However, the adoption of a coding or compression mechanism has important effects

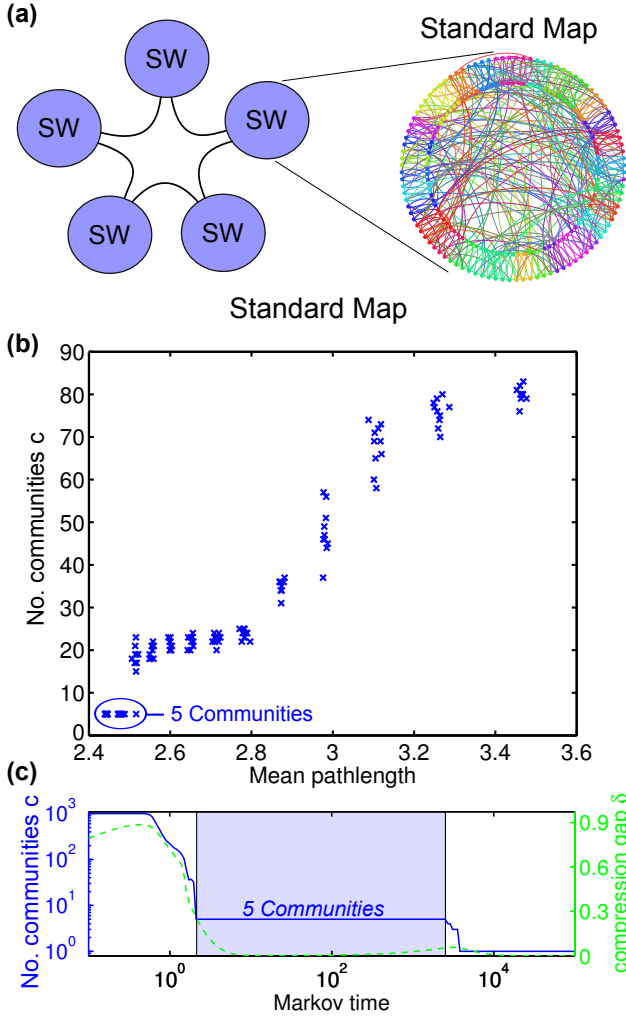


Figure 5. (Color online) **Community detection in a ring of small-world communities.** (a) Ring of 5 small-world communities with $N = 200$ each. The edges within the small-worlds have weight 5 while the weight of the links between them is 1. All the small-worlds have an average number of randomly added shortcuts per node, s . For $s = 1$ (shown), standard Map shows a strong overpartitioning leading to an average of 16 communities inside each small-world (indicated by different colors in online version). (b) Number of communities found by standard Map *vs.* mean pathlength inside the small-world communities. The numerics shown correspond to 10 different realizations of the network with average number of shortcuts per node, $s = 1, 1.25, 1.5, \dots, 3.75, 4$. (c) Applying Markov sweeping map to the ring of small-worlds with $s = 2.5$ (mean pathlength inside the small-worlds ≈ 2.7) finds the relevant partition into 5 communities, while standard Map finds 23 communities in this case.

on the outcome of the algorithm and ultimately reflects the underlying assumptions about the concept of community. Here we have shown that the original map equation formalism is inherently tuned towards a block-averaged notion of community structure as a weighted, statistical clique of cliques. This tuning stems from two inter-related simplifications: the block-averaged coding mech-

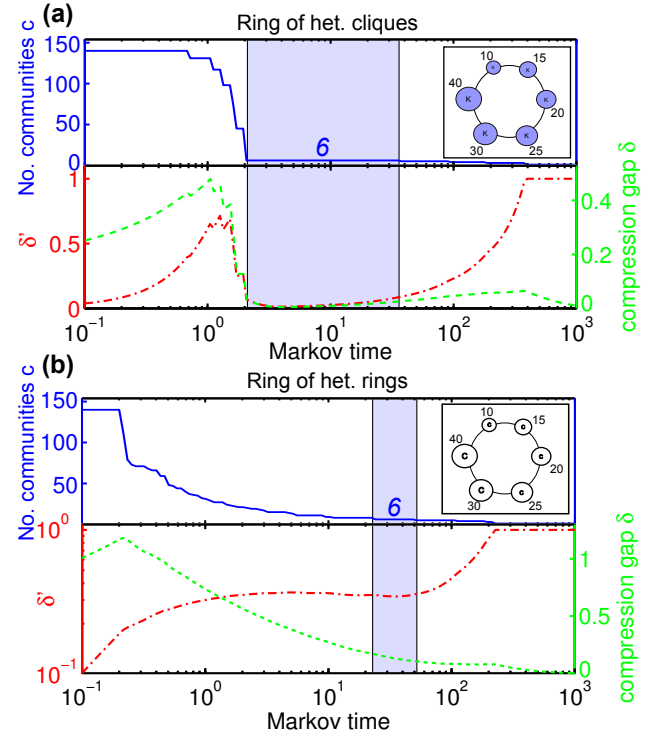


Figure 6. (Color online) **Community detection with heterogeneously sized subgraphs.** (a) Ring of 6 cliques with different sizes $\{10, 15, 20, 25, 30, 40\}$. Upper panel: number of communities found by Markov sweeping map *vs.* Markov time. Lower panel: both the compression gap δ (dashed green line) and the alternative compression gap measure δ' (red dashed-dotted line), clearly highlight the presence of a robust partition into 6 communities. Inset: analysed graph. (b) Ring of 6 rings with different sizes $\{10, 15, 20, 25, 30, 40\}$. In this case the alternative measure δ' for the compression gap is better suited for the analysis, indicating the presence of the 6 rings by a relative minimum around Markov time 30. Inset: analysed graph.

anism, which ignores the detailed connectivity and exhibits a large compression gap for non-clique structures, and the use of one-step quantities, which ignores the effect of multi-step flows in the communities and leads to an upper scale (field-of-view) for detection. This intrinsic bias of the map equation explains the excellent performance of the map equation in clique-like benchmarks but can lead to unexpected overpartitioning of networks if they differ strongly from the assumed clique-like organization.

We have shown that using the dynamical zooming provided by Markov time sweeping allows one to take into account multi-step flows and scan across all scales in a natural manner. The underlying idea is that, as time increases, the communities in the network will become more clique-like when analyzed through the time-dependent weighted transition matrix of the Markov process. Therefore, the map formalism can be used to detect long-range communities as the Markov time increases,

and the relevant communities will be signaled by a low compression gap. This Markov sweeping for the map equation can enhance the performance of the method by allowing it to detect non-clique communities and the presence of multi-scale community structure in networks. Importantly, the method still recovers all the results from the original map equation.

As stated above, the dynamic zooming across all scales effected by the Markov process is an integral ingredient of the method. Rather than just looking for the ‘right’ scale, the community structure emerges from the integration of the information gathered systematically at all scales. This approach can help alleviate the reliance on a global scale which can affect the results when dealing with networks with communities with very heterogeneous sizes [38]. In particular, Markov-sweeping map is able to detect heterogeneous cliques as obtained through the LFR benchmark, a fact consistent with the notion that cliques are all effectively one-step and that standard Map already performs effectively on such benchmarks (see also Figure 6 for an analysis with heterogeneously sized cliques). Similarly our method performs well in detecting communities in a ring of rings with very dissimilar sizes as illustrated in Figure 6, although when the heterogeneity of the relative ring sizes becomes very large, our approach will not identify all rings at once at the same level of the hierarchy. To improve further the applicability of the method to such problems, one can use different dynamics for the Markov process [19, 39]. This is an area of research we are currently pursuing. However, since there is no community detection algorithm that will serve all purposes for all possible applications, one should complement the analysis with other methods based on different principles (e.g., local algorithms in those cases).

Adding a dynamical dimension to Map through Markov sweeping is just one of the possible ways to en-

hance the map equation and alternative approaches are worth pursuing. One direction would be the modification of the coding scheme. For instance, a more rigorous treatment would require to remove the constraint of having unique codewords within each community and allow also for encoding of walks instead of single step codewords. This generalization, however, would most likely lead to a breakdown of the simple coding picture that underpins the map equation. Our work emphasizes the importance of the choice of dynamics on the network and shows that using a dynamical perspective may lead to a more natural framework for community detection, especially when the underlying system has an inherent flow. In this paper, we have used the standard (unbiased) continuous-time random walk as a neutral first choice of dynamics. However, other continuous time or discrete time processes are possible (see also [19] for a related discussion) in order to tune our community detection algorithm to different characteristics of the network.

Code is available online [40].

ACKNOWLEDGMENTS

We thank J.-C. Delvenne for fruitful discussions, especially relevant for the quantitative analysis of the rings. We thank S.N. Yaliraki for helpful comments. R.L. acknowledges funding from the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. M.B. acknowledges funding from grant EP/I017267/1 from the EPSRC (Engineering and Physical Sciences Research Council) of the UK under the Mathematics underpinning the Digital Economy program and from the US Office of Naval Research (ONR). The scientific responsibility rests with its authors.

-
- [1] S. Fortunato, *Physics Reports* **486**, 75 (2010).
 - [2] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, *Physics Reports* **424**, 175 (2006).
 - [3] A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou, *Physics Reports* **469**, 93 (2008).
 - [4] H. A. Simon, *Proceedings of the American Philosophical Society* **106**, 467 (1962).
 - [5] M. E. J. Newman and M. Girvan, *Phys. Rev. E* **69**, 026113 (2004).
 - [6] M. E. J. Newman, *Proceedings of the National Academy of Sciences* **103**, 8577 (2006).
 - [7] M. Fiedler, *Czechoslovak Mathematical Journal* **23**, 298 (1973).
 - [8] M. Fiedler, *Czechoslovak Mathematical Journal* **25**, 619 (1975).
 - [9] J. Shi and J. Malik, *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **22**, 888 (2000).
 - [10] R. Kannan, S. Vempala, and A. Veta, in *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on* (IEEE, 2000) pp. 367–377.
 - [11] M. Porter, J. Onnela, and P. Mucha, *Notices of the AMS* **56**, 1082 (2009).
 - [12] M. T. Schaub, J.-C. Delvenne, S. N. Yaliraki, and M. Barahona, *PLoS ONE* **7**, e32210 (2012).
 - [13] H. A. Simon and A. Ando, *Econometrica* **29**, 111 (1961).
 - [14] J. Reichardt and S. Bornholdt, *Phys. Rev. Lett.* **93**, 218701 (2004).
 - [15] A. Arenas, A. Fernández, and S. Gómez, *New Journal of Physics* **10**, 053039 (2008).
 - [16] P. Ronhovde and Z. Nussinov, *Phys. Rev. E* **80**, 016109 (2009).
 - [17] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, *PLoS ONE* **6**, e18961 (2011).
 - [18] J.-C. Delvenne, S. N. Yaliraki, and M. Barahona, *Proceedings of the National Academy of Sciences* **107**, 12755 (2010).
 - [19] R. Lambiotte, J.-C. Delvenne, and M. Barahona, (2009), [arXiv:0812.1770](https://arxiv.org/abs/0812.1770).
 - [20] E. Ziv, M. Middendorf, and C. H. Wiggins, *Phys. Rev. E* **71**, 046117 (2005).

- [21] M. Rosvall and C. T. Bergstrom, *Proceedings of the National Academy of Sciences* **105**, 1118 (2008).
- [22] M. Rosvall and C. T. Bergstrom, *PLoS ONE* **6**, e18209 (2011).
- [23] A. Raj and C. H. Wiggins, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**, 988 (2010).
- [24] A. Lancichinetti and S. Fortunato, *Phys. Rev. E* **80**, 056117 (2009).
- [25] S. Fortunato and M. Barthélemy, *Proceedings of the National Academy of Sciences* **104**, 36 (2007).
- [26] A. Viamontes Esquivel and M. Rosvall, *Phys. Rev. X* **1**, 021025 (2011).
- [27] C. E. Shannon, *The Bell System Technical Journal* **27**, 379 (1948).
- [28] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. (Wiley-Interscience, 2006).
- [29] Note that we do not have to construct a code to evaluate the map equation: one can find the optimal code-lengths from the entropy expressions or the map equation (3).
- [30] B. Karrer and M. E. J. Newman, *Phys. Rev. E* **83**, 016107 (2011).
- [31] A. Delmotte, E. W. Tate, S. N. Yaliraki, and M. Barahona, *Physical Biology* **8**, 055010 (2011).
- [32] More specifically, the standard map equation is exactly recovered at time $t = 1$ when considering either a linearized version of the dynamics (14) or of the corresponding discrete-time dynamics (1). The results presented here consider the nonlinear transition matrix (14) and hence the results at time $t = 1$ shown here can display small deviations from those obtained from the standard (i.e., linearized in our case) map scheme.
- [33] R. Lambiotte and M. Rosvall, *Phys. Rev. E* **85**, 056107 (2012).
- [34] D. J. Watts and S. H. Strogatz, *Nature* **393**, 440 (1998).
- [35] A. Taylor and D. J. Higham, *ACM Trans. Math. Softw.* **35**, 26:1 (2009).
- [36] M. E. J. Newman, C. Moore, and D. J. Watts, *Phys. Rev. Lett.* **84**, 3201 (2000).
- [37] M. Barahona and L. M. Pecora, *Phys. Rev. Lett.* **89**, 054101 (2002).
- [38] A. Lancichinetti and S. Fortunato, *Phys. Rev. E* **84**, 066122 (2011).
- [39] J.-C. Delvenne, M. T. Schaub, S. N. Yaliraki, and M. Barahona, in *Time Varying Dynamical Networks*, edited by N. Ganguly, A. Mukherjee, M. Choudhury, F. Peruani, and B. Mitra (Birkhäuser, Springer, Boston, 2012) to be published.
- [40] <http://michaelschaub.github.com/MarkovZoomingMap/>.